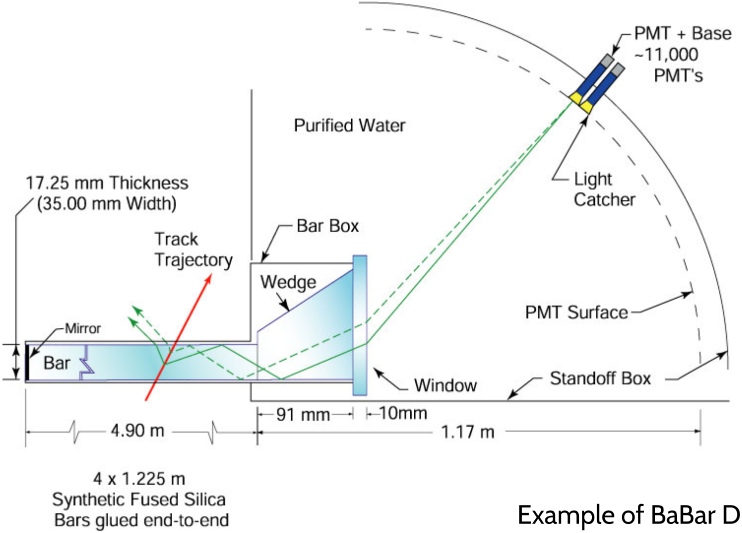
DIRC Pattern Recognition in High-Multiplicity

Wilka Carvalho, Axel Drees, Alan Dion

2014-12-17

Detection of Internally Reflected Cherenkov Light



Example of BaBar DIRC

How a DIRC identifies particles

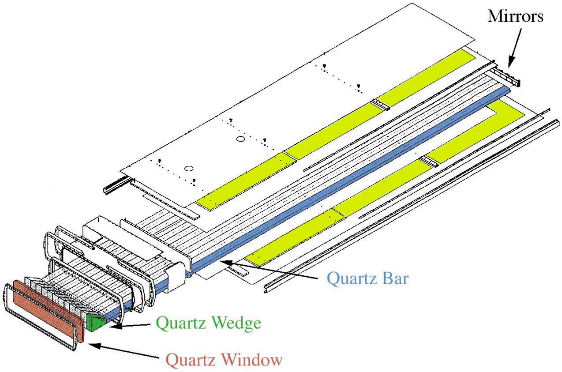
We have knowledge of an incoming charged track's $\mathbf{p}$ vector

If we knew the velocity of the particle, we would know the Cherenkov emission angle, and thus the expected pattern of photons coming out of the end of the bar (a rotated cone of light)

Inverting this knowledge, if we know the pattern of photons at the readout, we know the velocity. Since we also know the momentum, we know the mass.

Additionally, a DIRC readout can provide some timing information for the photons

Readout



In BaBar, many quartz bars are stacked together and readout into one giant expansion volume with pmt's at the end

Our Goals

There is a common wisdom that a DIRC will not work in a high-multiplicity environment

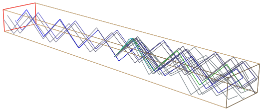
But what if each bar could be read out separately? In a central heavy ion collision, there would be a few 10s of charged particles traversing each bar. Can pattern recognition sort out the PID?

Axel, myself, and Wilka Carvalho (SBU undergrad) are approaching this problem form a proof-of-principle standpoint for now :

assume we have focusing optics
assume we have an array of micro-pmts with ~single photon readout
what multiplicities could we deal with in principle?

Monte Carlo generators

We are using two options to simulate Cherenkov light in a DIRC bar. One is from pandaroot (experiment at FAIR)



The other is a toy mc which allows us to play in a simple environment:
no frequency-dependence of index-of-refraction
all resolutions are just put into one simple single-photon resolution

Actual BaBar photon resolution is about 10 mrad. This is from chromatic smearing, imaging granularity and propagation. I expect the μPMT readout would ultimately limit us to more like 30 mrad, but for now we use 10 mrad in our studies.

Monte Carlo generators

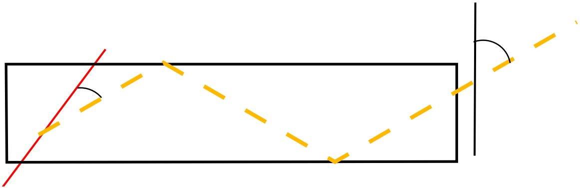
Toy generator :
`https://code.google.com/p/particle-detector/`

README at
`https://code.google.com/p/particle-detector/source/browse/FULLRUN_README`

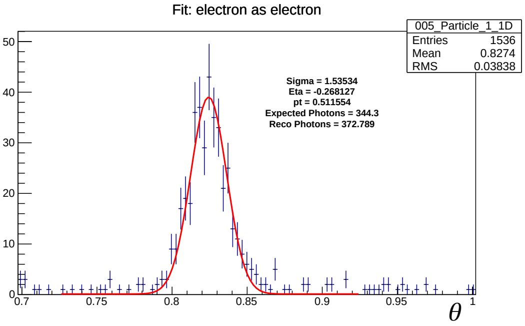
PandaROOT :
`http://fairroot.gsi.de/?q=node/7`

BOTH packages are tricky to install.

Basic pattern recognition

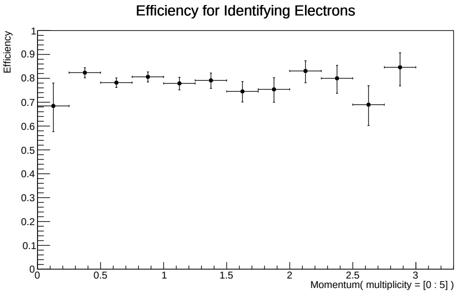


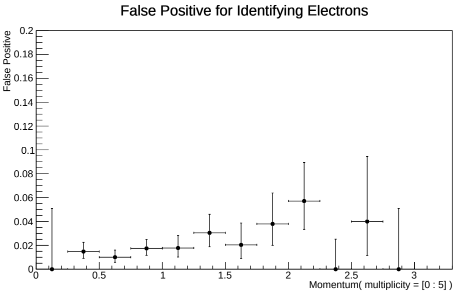
For each charged track, rotate all of the measured photon vectors into the polar frame of that track. Project the light onto the polar angle axis and plot it :



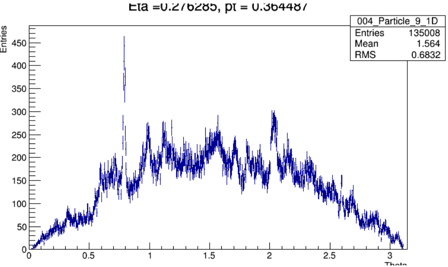
Single Track performance

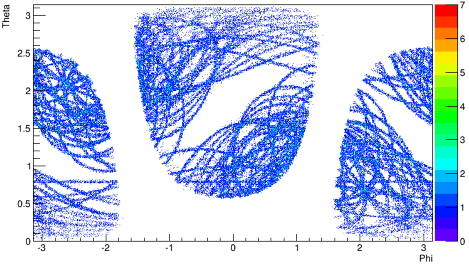
For each track, we guess that a particle is either a pion,kaon,proton,electron, or muon. We compare the reconstructed Cherenkov angle to the expected angle, and the number of photons in the peak as well. We make a cut on the number of standard deviations from the expectation in angle and yield





The left is the efficiency for electrons with a 1.5σ cut. The right is the rate at which pions are identified as electrons. This is close to what BaBar sees in reality.





It is encouraging that one can see a peak even with such a rudimentary procedure! If we really would have 10 mrad resolution then we could likely do PID with just the projections on the left.

The 2D histogram has a lot of information in it when combined with the tracking.

What we are working on now

Moving over to pandaroot (which we have running) and maybe a bit later GEANT (which we don't)

Validate the performance of the simple algorithm described above as a function of multiplicity and single-photon resolution

The big project : more sophisticated pattern recognition
We can iteratively subtract photons from the (θ, ϕ) histogram before fitting in only θ-space
Also we can fit in (θ, ϕ) simulataneously

The goal is to get a NIM paper on the algorithm by the summer